Appendix 1

```
        void sw ( )
        (

5

        #define iw = 12;                        /* instruction
                                                width */
        #define mw = 3:                         /* memory width */
        #define CONST = 0                       /* push constant */
10      #define LOAD = 1                        /* push variable */
        #define GLOBAL = 2                       /* push address */
        #define PUTCHAR, = 15 /*                put a character along the
                                                standard output channel*/
        #define GETCHAR = 16 /*                 get a character from the
15                                              standard input channel */


        ...


        rom program []
20      #include "prog.o" ): ram stack[1«mw] with dualport = 1 ];
        ram memory[1«mw] unsigned iw PC, ir, tos;
        unsigned mw sp;


        do par it = program[pc]: PC = PC + 1;
25      tos = stack[sp-1];                          /* save top of
                                                    stack to avoid
                                                    two ram accesses
                                                    in one cycle
                                                    */

30

        switch (ir)
```

```
case
CONST par
        stack[sp] = program[pc];
        sP = sP+l:
        PC = Pc+l:
        ]
        break;
case LOAD
        stack[sp-1] = memory[tos<-mw];
        break;
case STOP break; default :                    /* unknown opcode */
while (1) delay;

] while (ir != STOP);
]
```

Register transfer level description of simple processor